CLAIM(S):

1. A method for providing computer application security, the method comprising:

identifying secured resources within a software application;

grouping secured resources into user roles in a data store;

creating a plurality of surrogate identifiers in the data store, each surrogate identifier being associated with one user role;

associating users with user roles, each user being associated with one user role; and

determining access rights to the secured resources for each user according to a corresponding surrogate identifier without disclosing the corresponding surrogate identifier to the user, the corresponding surrogate identifier being associated with the one user role of the user.

2. The method of claim 1, wherein identifying secured resources comprises:

identifying functions within the software application to be secured, the identified functions being secured resources; and

invoking a security call before permitting access to the secured resources.

3. The method of claim 2, wherein identifying secured resources further comprises:

installing an embedded module in the software application to capture the security call.

4. The method of claim 1, wherein grouping secured resources into user roles comprises:

establishing in the data store links to each of the secured resources;
selecting the links corresponding to related secured resources;
grouping the selected links into user roles; and
storing the user roles in the data store.

5. The method of claim 1, wherein grouping secured resources into user roles comprises:

establishing in the data store links to each of the secured resources within the software application;
selecting the links corresponding to related secured resources;
grouping the selected links into privilege sets;
grouping privilege sets and links into user roles; and
storing the user roles in the data store.

6. The method of claim 1, wherein grouping secured resources into user roles comprises:

establishing in the data store links to each of the secured resources within the software application;
selecting the links corresponding to related secured resources;
grouping the selected links into privilege sets;
grouping privilege sets and links into job functions;
grouping job functions, privilege sets and links into user roles; and
storing the user roles in the data store.

7. The method of claim 1, wherein creating a plurality of surrogate identifiers comprises:

associating each surrogate identifier with one user role in the data store; and
replicating each surrogate identifier in a data store of a security provider.

8. The method of claim 1, wherein associating a user with a user role comprises:

creating a list of user identifiers corresponding to existing users on a security provider;

selecting user identifiers from the list;

storing selected user identifiers in the data store; and

associating each selected user identifier with one user role, the user role being undisclosed to the user.

9. The method of claim 1, wherein determining access rights to one of the secured resources comprises:

authenticating the user as a valid user; and

authorizing the user to access one of the secured resources.

10. The method of claim 9, wherein authenticating the user comprises:

invoking programatically an embedded component within the software application when a secured resource is accessed;

passing a resource name identifying the secured resource through the embedded component to a platform coordinator;

retrieving an identifier and a security provider name from the user via the platform coordinator;

passing the identifier and the security provider name to a security broker;

relaying the identifier to a security provider associated with the security provider name for authentication;

evaluating automatically the identifier against a data store of the security provider;

returning an authentication result to the security broker;

storing an authentication token with a time stamp in a cache of the security broker when authentication is successful, the

authentication token created by the security broker based on the authentication result;

retrieving the user role associated with the identifier from the data store;

retrieving the surrogate identifier associated with the user role from the data store;

passing the surrogate identifier and a secured resource name from the security broker to the security provider;

evaluating automatically the surrogate identifier against the data store of the security provider;

determining automatically permissions associated with the surrogate identifier on the security provider;

returning an authorization result associated with the surrogate identifier to the security broker;

creating automatically a permissions token on the security broker based on the authorization result;

relaying the permissions token to the platform coordinator, the permissions token comprising both the secured resource and access rights;

storing the permissions token with a time stamp in a cache on the platform coordinator; and

relaying the access rights to the software application through the embedded component.

11. The method of claim 9, wherein once the user is authenticated, authorizing the user comprises:

invoking programatically an embedded component within the software application when a secured resource is accessed;

passing a resource name identifying the secured resource through the embedded component to a platform coordinator;

retrieving an authentication token from a cache on the platform coordinator;

passing the authentication token and the resource name to the security broker;

comparing the authentication token against the cache on the security broker to identify a matching authentication token, the matching authentication token being associated in the cache with the surrogate identifier;

passing the surrogate identifier and the resource name from the security broker to the security provider;

evaluating automatically the surrogate identifier against the data store of the security provider;

determining automatically permissions associated with the surrogate identifier on the security provider;

returning an authorization result associated with the surrogate identifier to the security broker;

creating automatically a permissions token on the security broker based on the authorization result;

relaying the permissions token to the platform coordinator, the permissions token comprising both the secured resource and access rights;

storing the permissions token with a time stamp in a cache on the platform coordinator; and

relaying the access rights to the software application through the embedded component.

12. The method of claim 9, wherein once the user is authenticated and authorized to access the secured resource, determining access rights to one of the secured resources further comprises:

invoking programatically an embedded component within the software application when the secured resource is accessed;

passing a resource name identifying the secured resource through the embedded component to a platform coordinator;

retrieving an authentication token from a cache on the platform coordinator;

comparing the secured resource name with permissions tokens stored in the cache on the platform coordinator for a matching permissions token, the matching permissions token containing the secured resource name;

relaying access rights associated with the matching permissions token to the software application through the embedded component.

13. A method for providing computer security, the method comprising:

securing a plurality of resources within a software application;

identifying each of the plurality of resources in a data store;

selecting some of the plurality of resources;

grouping selected resources into user roles in the data store;

creating a plurality of user names and a plurality of aliases in the data store, each user name and each alias being associated with the same one user role;

replicating the plurality of resources, the user roles, the plurality of user names and the plurality of aliases in a plurality of data stores; and

determining access privileges to the plurality of resources using an alias corresponding to a user name by virtue of the same one user role from one of the plurality of data stores.

14. The method for providing computer security of claim 13, wherein determining access privileges comprises:

authenticating a user on the system; and

authorizing access rights to secured resources in the software application.

15. The method for providing computer security of claim 14, wherein authenticating a user comprises:

retrieving a user identifier;

passing the user identifier to a security provider;

verifying the user identifier against a data store on the security provider; and

returning an encrypted authentication token.

16. The method for providing computer security of claim 14, wherein authorizing access rights comprises:

capturing a security call from the software application, the security call containing a name identifying a secured resource;

retrieving a user identifier;

passing the user identifier to a security broker;

retrieving one of the plurality of aliases from the data store of the security broker, the retrieved alias corresponding to the user identifier;

passing the retrieved alias to a security provider;

verifying the alias against a provider data store on the security provider;

returning an encrypted permissions token to the software application; and

determining access rights to the secured resource according to the permissions token.

17. The method of claim 16 wherein retrieving a user identifier comprises:

gathering information about a user for authorizing access to secured resources, the information selected from the group consisting of user name and password, software token, hardware token, and digital signature.

18. A computer security system comprising:

a plurality of computer workstations, each computer workstation having an operating system and a software application installed, the software application containing an embedded component;

a plurality of security providers, each security provider having a security data store; and

a plurality of security brokers, each security broker having a data store, each security broker being a computer in network communication with the computer workstations and the security providers;

wherein each computer workstation is capable of communicating with each security broker; and

wherein each security broker is capable of communicating with each security provider.

19. The computer security system of claim 18, wherein the computer workstations further comprise:

a platform coordinator installed on each workstation, the platform coordinator for routing permissions requests to security brokers, the platform coordinator capable of communicating with any one of the security brokers so that if one of the security brokers is unavailable, the platform coordinator can route the permissions requests to another security broker for proceeding with authentication and authorization.

20. The computer security system of claim 18, wherein the security brokers further comprise:

a cache for storing an authentication token, the authentication token being used to retrieve a surrogate identifier associated with the authentication token.

21. The computer security system of claim 18, wherein the security brokers route permissions requests programmatically to the security providers, each security broker being capable of routing permissions requests to any one of the security providers such that if one security provider is unavailable, the security broker can route permissions requests to another security provider.

22. The computer security system of claim 18, wherein the security system further comprises:

administration utilities for configuring, updating and maintaining the data store and the security data store, the administration utilities providing a single software application for maintaining user identifiers, setting and changing permissions, creating security events, and tracking system usage and security events within the security system.

23. A process for authorizing access rights to secured resources in a software application, the process comprising:

authenticating a computer user to a computer security provider via a user identifier corresponding to the computer user, the computer security provider returning a result to a security broker according to the user identifier;

storing the result on the security broker;

retrieving a surrogate identifier from the security broker, the surrogate identifier corresponding to the result, the surrogate identifier being undisclosed to the computer user; and

authorizing the surrogate identifier to the computer security provider, the computer security provider returning surrogate permissions to the security broker, the surrogate permissions corresponding to the surrogate identifier, the surrogate permissions for determining access rights to secured resources in the software application according to the surrogate permissions.

24. The process for authorizing access rights according to claim 23, wherein authorizing the surrogate identifier to the computer security provider comprises:

passing the surrogate identifier to a security manager;

querying for the surrogate identifier in a permissions list on the security provider using the security manager;

determining surrogate permissions for the surrogate identifier according to the permissions list; and

returning the surrogate permissions to the security broker.

25. The process for authorizing access rights according to claim 24, wherein authorizing the surrogate identifier to the computer security provider further comprises:

passing the surrogate permissions from the security broker to a platform coordinator;

storing the surrogate permissions with a time stamp in a cache on the platform coordinator;

relaying the surrogate permissions to an embedded component within the software application;

passing the surrogate permissions to a function within the software application, the function capable of interpreting the surrogate permission; and

interpreting the surrogate permission using the function to permit or deny access rights to the secured resource.

26. The process for authorizing access rights according to claim 23, wherein authenticating comprises:

passing the user identifier from the security broker to a security manager;

querying for the user identifier in an authentication list on the security provider using the security manager;

determining validity of the user identifier according to the authentication list; and

returning a result to the security broker.